

In the claims:

For the Examiner's convenience, all pending claims are presented below with changes shown in accordance with the mandatory amendment format.

1. (Currently Amended) A method comprising:

executing a speculative read-reordered load instruction prior to a potentially conflicting load in an instruction sequence;

storing memory conflict information representing the speculative read-reordered load;

executing a read-reordered load check instruction associated with the speculative read-reordered load instruction, the read-reordered load check instruction to determine if
matching an address of [[a]] the potentially conflicting load matches against an address of
the stored memory conflict information; and

invalidating the stored memory conflict information with a matching address to
the address of the potentially conflicting load if the stored memory conflict information
has a value different than a value of the potentially conflicting load.

2. (Currently Amended) The method of claim 1, ~~wherein~~ further comprising

validating the stored memory conflict information with the matching address is

~~invalidated~~ if the stored memory conflict information has the same ~~different~~ value as the
value of ~~than~~ the potentially conflicting load.

3. (Currently Amended) The method of claim 2, ~~further comprising executing a~~

wherein the validating of the stored memory conflict information of the read re-ordered

load check instruction to determine the validity of the speculative read re-ordered load

further comprises passing control by the read re-ordered load check instruction to a next instruction in the instruction sequence.

4. (Original) The method of claim 1, wherein the memory conflict information is stored in a read re-ordered load address table (RRLAT).

5. (Currently Amended) The method of claim 4 [[5]], further comprising updating the stored memory conflict information by setting a validity bit in the RRLAT to a valid state when new memory conflict information is stored.

6. (Currently Amended) The method of claim 5 [[6]], further comprising setting the validity bit to an invalid state if a later conflicting load operation is executed.

7. (Currently Amended) A processor, comprising:

a re-ordered load address table (RRLAT) to store memory conflict information representing a speculative read re-ordered load that is executed prior to a potentially conflicting load in an instruction sequence; and

a monitor to:

compare a potentially conflicting load against the stored memory conflict information by executing a read-reordered load check instruction associated with the speculative read-reordered load instruction, the read-reordered load check instruction to determine if an address of the potentially conflicting load matches an address of the stored memory conflict information; and

invalidate the stored memory conflict information with a matching address to the address of the potentially conflicting load if the stored memory conflict information has a value different than a value of the potentially conflicting load.

8. (Currently Amended) The processor of claim 7 [[8]], wherein the stored memory conflict information with the matching address is [[in]]validated if the stored memory conflict has ~~a matching address~~ the same value as the value of the potentially conflicting load.

9. (Currently Amended) The processor of claim 8, wherein the validating of the stored memory conflict information of the read re-ordered load check instruction further includes ~~is invalidated if the stored memory conflict has a matching address and a different value than the potentially conflicting load~~ passing control by the read re-ordered load check instruction to a next instruction in the instruction sequence.

10. (Currently Amended) The processor of claim 7 [[10]], wherein the RRLAT is referenced upon the execution of [[a]] the read re-ordered load check instruction to determine the validity of the speculative read re-ordered load.

11. (Currently Amended) The processor of claim 7 [[8]], wherein the RRLAT may be any one of a direct-mapped, multi-way set associative, and fully associative data structure.

12. (Currently Amended) The processor of claim 7 [[8]], wherein the RRLAT is portioned among hardware thread contexts.

13. (Currently Amended) The processor of claim 7 [[8]], wherein the RRLAT includes storage locations for an address, a target register ID, a value, and validity information associated with the speculative read re-ordered load.

14.-15. (Cancelled)

16. (Currently Amended) A computer system, comprising:

a first processor; and

a second processor, including:

a re-ordered load address table (RRLAT) to store memory conflict information representing a speculative read re-ordered load received from the second processor that is executed prior to a potentially conflicting load in an instruction sequence; and

a monitor to:

compare a potentially conflicting load received from the first processor against the stored memory conflict information by executing a read-reordered load check instruction associated with the speculative read-reordered load instruction, the read-reordered load check instruction to determine if an address of the potentially conflicting load matches an address of the stored memory conflict information; and ~~and to~~

invalidate the stored memory conflict information if the stored memory conflict information has a matching address and a different value than the potentially conflicting load.

17. (Currently Amended) The computer system of claim 16 [[19]], wherein the stored

memory conflict information with the matching address is validated if the stored memory conflict has the same value as the value of the potentially conflicting load~~monitor unit~~
~~executes a read re-ordered load check instruction to determine the validity of the~~
~~speculative read re-ordered load.~~

18. (Currently Amended) A computer system, comprising:

a memory device; and

a processor coupled to the memory device, including:

a re-ordered load address table (RRLAT) to store memory conflict
information representing a speculative read re-ordered load that is executed prior
to a potentially conflicting load in an instruction sequence;

a monitor to:

compare a potentially conflicting load against the stored memory conflict
information by executing a read-reordered load check instruction associated
with the speculative read-reordered load instruction, the read-reordered load
check instruction to determine if an address of the potentially conflicting load
matches an address of the stored memory conflict information; and[[, to]]

invalidate the stored memory conflict information if the stored memory
conflict information has a matching address and a different value than the
potentially conflicting load; and

a cache memory;~~and~~

~~a memory device coupled to the processor.~~

19. (Currently Amended) The computer system of claim 18 [[21]], wherein the

monitor unit validates the stored memory conflict information with ~~[[a]]~~ the matching address if the stored memory conflict information has ~~a matching~~ the same value as the value of the potentially conflicting load.

20. (Currently Amended) The computer system of claim 19 ~~[[22]]~~, wherein the validating of the stored memory conflict information of the read re-ordered load check instruction further includes passing control by the read re-ordered load check instruction to a next instruction in the instruction sequence ~~monitor unit executes a read re-ordered load check instruction to test the validity of the speculative read re-ordered load.~~

21. (Currently Amended) The computer system of claim 18 ~~[[21]]~~, further comprising a bus to control communications between the processor and the memory device.

22. (Currently Amended) A machine-readable medium storing a sequence of instructions that, when executed by a machine, cause the machine to:

execute a speculative read-reordered load instruction prior to a potentially conflicting load in an instruction sequence;

store memory conflict information representing the speculative read-reordered load;

execute a read-reordered load check instruction associated with the speculative read-reordered load instruction, the read-reordered load check instruction to determine if an ~~match the~~ address of ~~[[a]]~~ the potentially conflicting load matches ~~against~~ an address of the stored memory conflict information; and

invalidate stored memory conflict information with a matching address if the stored memory conflict information has a different value than a value of the potentially conflicting load.

23. (Currently Amended) The machine-readable medium of claim 22 ~~[[25]]~~, the sequence of instructions, when executed by the computer system, further causing the computer system to validate stored memory conflict information with a matching address if the stored memory conflict information has the same ~~a matching~~ value as the value of ~~[[to]]~~ the potentially conflicting load.

24. (Currently Amended) The machine-readable medium of claim 23 ~~[[26]]~~, wherein the validating of the stored memory conflict information of the read re-ordered load check instruction further comprises passing control by the read re-ordered load check instruction to a next instruction in the instruction sequence ~~the sequence of instructions, when executed by the computer system, further causing the computer system to execute a read re-ordered load check instruction to determine the validity of the speculative read re-ordered load.~~